

2340

NRL Memorandum Report 2340
NRL Computer Bulletin 25

A Fortran Subroutine to Produce Combinations of n Distinct Things Taken m at a Time

JAMES P. GRIMES

*Research Computation Center
Mathematics and Information Sciences Division*

July 1971



NAVAL RESEARCH LABORATORY
Washington, D.C.

Approved for public release; distribution unlimited.

CONTENTS

Abstract	ii
Problem Status	ii
Authorization	ii
1. IDENTIFICATION	1
2. PURPOSE	2
3. USAGE	3
4. METHOD OR ALGORITHM	6
5. SOURCE LANGUAGE LISTING	7
6. COMPARISON	12
7. TEST METHOD AND RESULTS	12

ABSTRACT

A Fortran subroutine to generate all or particular combinations of n things taken m at a time is herein described. The procedure establishes and tabulates a certain one-to-one correspondence between the integers from 1 to ${}_nC_m$ and the individual combinations of n distinct "things" taken m at a time, where

$${}_nC_m = \frac{n!}{m!(n-m)!}$$

In all cases, the set of "things" treated by the subroutine is the set of integers $\{1, \dots, n\}$.

PROBLEM STATUS

This is a final report on this phase of the problem.

AUTHORIZATION

NRL General and Administrative Function 78-1601

A Fortran Subroutine to Produce Combinations
of n Distinct Things Taken m at a Time

1.0 IDENTIFICATION

1.1 Title

Combinations Subroutine

1.2 Identification Name

GO-NRL-CMBN

1.3 Classification Code

GO - Statistical Analysis and Probability, General

1.4 RCC Identification Name

G0001R00

1.5 Entry Points

CMBN

1.6 Programming Language

Language: 3600/3800 FØRTRAN

Routine Type: Subroutine

Operating System: Drum Scope 2.0

1.7 Computer and Configuration

CDC 3800

1.8 Contributor or Programmer

James P. Grimes, Code 7816JG, Research Computation
Center, Mathematics and Information Sciences
Division

1.9 Contributing Organization

NRL - Naval Research Laboratory, Washington, D. C.
20390

1.10 Program Availability

1.10.1 Submittal: Program write-up, Fortran source deck, source listing

1.10.2 On File: RCC Program Library

1.11 Verification

The subroutine was checked for the following values of N, M, and NN; (see Section 3.2 for definition of N, M, and NN).

<u>N</u>	<u>M</u>	<u>NN</u>
17	15	0
17	16	0
20	20	0
101	100	2
0	0	1
20	20	1
19	0	0
1	1	0
1	1	1
6	7	0
0	1	0

1.12 Date

1 July 1971

2.0 PURPOSE

2.1 Description of the Routine

To generate all or particular combinations of the integers from 1 to n taken a given number at a time.

This procedure establishes and tabulates a certain one-to-one correspondence between the integers from 1 to n and the individual combinations of n distinct "things" taken m at a time. Here

$${}_nC_m = \frac{n!}{m!(n-m)!}$$

In all cases, the set of "things" treated by the subroutine is the set of integers {1,...,n}.

The type of correspondence referred to may be illustrated for the case $n=5$, and $m=3$, as follows:

Index	Combination
1	1,2,3
2	1,2,4
3	1,2,5
4	1,3,4
5	1,3,5
6	1,4,5
7	2,3,4
8	2,3,5
9	2,4,5
10	3,4,5

2.2 Problem Background

In many phases of investigative mathematics, statistics, probability and related areas, the need often arises to generate and tabulate all or particular combinations of n distinct "things" taken m at a time. A computerized algorithm is the only practical way to handle this problem, especially as the number of combinations and the number to be taken at a time become relatively large. Previously, this type of computerized procedure was unavailable to NRL's scientific community.

3.0 USAGE

3.1 Calling Sequence or Operational Procedure

The calling sequence for the subroutine is:
 CALL CMBN (N, M, NN, MM)
 All four parameters are in CDC 3600/3800 integer format.

3.2 Arguments, Parameters, and/or Initial Conditions

Input parameters:

N and M are the n and m described in Section 2.1, i.e., as in ${}_nC_m$.

NN is an input integer which establishes the correspondence referred to in Section 2.1.

If $NN=0$, the correspondence is set by the subroutine and all combinations from 1 to ${}_nC_m$ are generated,

indexed, and printed out by the subroutine. Otherwise, if $NN = k$, $k = 1, \dots, {}_nC_m$, only that combination which corresponds to k is generated, indexed, and printed out as demonstrated in Section 2.1. MM is a one-dimensional input array which must be dimensioned in the calling program and whose length must be at least $4(m+1)$.

Output parameters:

There are no output parameters as such since all output is performed by the subroutine. However, after execution of the subroutine for each value of NN, whether supplied by the user or the subroutine itself, the m integers comprising the output combination occupy the first m locations of the array MM. The subroutine prints out this combination and the MM array is reused for the next combination. This procedure is repeated until all combinations ($NN = 0$) or the particular combination ($NN = k$, $k = 1, \dots, {}_nC_m$) is generated and printed out.

3.2.1 Special Provisions and Restrictions

Apparently, the basic algorithm imposes no restrictions on the parameters. However, there are computer-imposed limitations, such as storage capacity, word size, etc. Nevertheless, to facilitate the output, n , m , and ${}_nC_m$ have been limited to 10^5 . Of course, $n \geq m$.

For those cases in which $m \geq 16$, CMBN uses logarithms for some of the intermediate steps; this fact results in appreciably longer computation time for these cases. For the most part, computation time depends on the size of m and ${}_nC_m$.

3.3 Space Required (Decimal and Octal)

3.3.1 Unique Storage: 1475 octal (829 decimal) locations in CMBN

The dimension of the input array MM is user supplied.

3.3.2 Common Blocks: None

3.3.3 Temporary Storage: None

3.4 Messages and Instructions to the Operator

None

3.5 Error Returns, Messages, and Codes

None

3.6 Informative Messages to the User

The message "INVALID INPUT PARAMETER(S) $N, M \leq 0$ OR $M > N$ " will be printed, if the input parameters are invalid.

3.7 Input

See the "Input parameters" described in Section 3.2

3.8 Output

See "Output parameters" in Section 3.2

3.9 Formats

None

3.10 External Routines and Symbols

None

3.11 Timing

Running time is primarily dependent on the size of m and $n C_m$. Some examples of running time on the CDC-3800 computer follow: $7 C_4$, 24 seconds; $9 C_3$, 24 seconds; $15 C_5$, 72 seconds; $20 C_{17}$, 91 seconds.

3.12 Accuracy

The procedure has been tested by the author for values of m up to 100. Apparently, it continues to be valid for higher values of m , but computation time greatly increases.

3.13 Cautions to Users

3.13.1 MM is an input array and must be dimensioned in the main program. Its size must be at least $4(m+1)$. The array name in the main program may be any integer variable.

- 3.13.2 Since the value of $n C_m$ can become very large, the user should make sure he does not exceed storage capacity or machine word size using integer format. To facilitate the output by the subroutine n , m , and $n C_m$ must be less than 10^5 .
- 3.13.3 The user should consider running time and the amount of output produced for large values of m and $n C_m$.
- 3.13.4 Of course, n and m must be positive and $n \geq m$.

3.14 Program Deck Structure

⁷JØB card
⁹

⁷FTN card
⁹

main program
CMBN subroutine

SCØPE

⁷LØAD card
⁹

⁷RUN card
⁹

⁷⁷(end-of-file card)
⁸⁸

3.15 References - Literature - Appendices

Narec Bulletin #120, Neliac Bulletin #58, by Peter C. Ryan, formerly Code 6410, NRL.

4.0 METHOD OR ALGORITHM

The procedure used in the subroutine is a modification and generalization of an algorithm used in a NELIAC-N computer function by Peter C. Ryan, formerly Code 6410, NRL. For further information or details, see reference listed in Section 3.15.

5.0 SOURCE LANGUAGE LISTING

	SUBROUTINE CMBN(N,M,NN,MM)	000
C	G0=NRL-CMBN,CDC 3800 COMPUTER SUBROUTINE TO PRODUCE	100
C	COMBINATIONS OF N DISTINCT THINGS TAKEN M AT A TIME.	200
C	(IN ALL CASES, THE SET OF THINGS TREATED BY THE	300
C	SUBROUTINE IS THE SET OF INTEGERS %1,...,NE)	400
C	TITLE=COMBINATIONS SUBROUTINE	500
C	IDENT NAME=G0-NRL-CMBN	600
C	LANGUAGE=FORTRAN	700
C	PROGRAMMER=JAMES P. GRIMES, CODE 7816JG	800
C	RESEARCH COMPUTATION CENTER, MISO	900
C	(A MODIFICATION AND GENERALIZATION OF A	1000
C	NELIAC=N COMPUTER FUNCTION BY MR.	1100
C	PETER C. RYAN, FORMERLY CODE 6410, NRL)	1200
C	IDENT NUMBER=G0001R00	1300
C	ORGANIZATION=NRL-NAVAL RESEARCH LABORATORY	1400
C	WASHINGTON, D.C. 20390	1500
C	DATE=2 NOVEMBER 1970	1600
C	PURPOSE=TO GENERATE ALL OR PARTICULAR COMBINATIONS	1700
C	OF THE INTEGERS FROM 1 TO N.	1800
C		1900
C	USAGE=THE CALLING SEQUENCE FOR THE SUBROUTINE IS:	2000
C	CALL CMBN(N,M,NN,MM)	2100
C	ALL FOUR PARAMETERS ARE IN CDC 3600/3800	2200
C	INTEGER FORMAT.	2300
C		2400
C	INPUT PARAMETERS=N AND M ARE THE INPUT INTEGERS IN	2500
C	C(N,M), THAT IS, THE COMBINATIONS	2600
C	OF N THINGS TAKEN M AT A TIME.	2700
C		2800
C	NN, AN INPUT INTEGER WHICH ESTABLISHES	2900
C	A CERTAIN ONE-TO-ONE CORRESPONDENCE	3000
C	BETWEEN THE INTEGERS FROM 1 TO C(N,M)	3100
C	AND THE INDIVIDUAL COMBINATIONS OF N	3200
C	THINGS TAKEN M AT A TIME.	3300
C		3400
C	IF NN=0, THE CORRESPONDENCE IS SET BY	3500
C	THE SUBROUTINE AND ALL COMBINATIONS	3600
C	FROM 1 TO C(N,M) ARE GENERATED,	3700
C	INDEXED, AND PRINTED OUT BY THE	3800
C	SUBROUTINE.	3900
C		4000
C	IF NN=K, K=1,...,C(N,M), ONLY THAT	4100
C	COMBINATION WHICH CORRESPONDS TO K	4200
C	IS GENERATED, INDEXED, AND PRINTED	4300
C	OUT.	4400
C		4500
C	MM IS A ONE-DIMENSIONAL INPUT ARRAY	4600
C	WHICH MUST BE DIMENSIONED IN THE	4700
C	CALLING PROGRAM AND WHOSE LENGTH	4800
C	MUST BE AT LEAST 4*(M+1).	4900

5000
5100
5200
5300
5400
5500
5600
5700
5800
5900
6000
6100
6200
6300
6400
6500
6600
6700
6800
6900
7000
7100
7110
7120
7130
7140
7150
7200
7300
7400
7500
7600
7700
7800
7900
8000
8100
8200
8300
8400
8500
8600
8700
8800
8900
9000
9100
9200
9300
9400

NN=II	9500
14 MN=4*(M+1)	9600
DO 15,I=1,MN	9700
15 MM(I)=0	9800
IM=M+1	9900
IC=IM+M+1	10000
IN=IC+M+1	10100
IF(M.EQ.1)16,17	10200
16 MM=NN	10300
GO TO 45	10400
17 MM(IC+1)=N	10500
MM(IN+1)=M	10600
IL=0	10700
IF(M.GT.15)30,18	10800
18 M1=M-1	10900
DO 29,J=1,M1	11000
MM(IC+J+1)=MM(IC+J)+MM(IN+J)+1	11100
IA=MM(IC+J+1)	11200
IB=1	11300
IH=IL	11400
MJ=M-J	11500
DO 19,L=1,MJ	11600
19 IB=IB*L	11700
DO 26,I=1,IA	11800
ID=1	11900
DO 20,L=1,MJ	12000
20 ID=ID*(IA-I+L)	12100
IE=ID/IB	12200
IL=IH	12300
IH=IL+IE	12400
IF(IH.GT.NN)21,22	12500
21 MM(IN+J+1)=I	12600
GO TO 27	12700
22 IF(NN.EQ.IH)23,26	12800
23 MM(IN+J+1)=I	12900
IF=0	13000
DO 24,L=1,J	13100
24 IF=IF+MM(IN+L+1)	13200
MM(IN+J+2)=N-M-IF+J+1	13300
J2=J+2	13400
DO 25,L=J2,M	13500
25 MM(IN+L+1)=1	13600
GO TO 42	13700
26 CONTINUE	13800
27 IF(M-1.EQ.J)28,29	13900
28 MM(IN+M+1)=NN-IL	14000
29 CONTINUE	14100
GO TO 42	14200
30 M1=M-1	14300
DO 41,J=1,M1	14400

MM(IC+J+1)=MM(IC+J)-MM(IN+J)+1	14500
IA=MM(IC+J+1)	14600
FB=0	14700
IH=IL	14800
MJ=M-J	14900
DO 31,L=1,MJ	15000
FL=L	15100
FL=LOGF(FL)	15200
31 FB=FB+FL	15300
DO 38,I=1,IA	15400
FD=0	15500
DO 32,L=1,MJ	15600
ID=IA-I+L	15700
FA=ID	15800
FA=LOGF(FA)	15900
32 FD=FD+FA	16000
FD=FD-FB	16100
FD=EXP(FD)+.5	16200
IE=FD	16300
IL=IH	16400
IH=IL+IE	16500
IF(IH.GT.NN)33,34	16600
33 MM(IN+J+1)=I	16700
GO TO 39	16800
34 IF(NN.EQ.IH)35,38	16900
35 MM(IN+J+1)=I	17000
IF=0	17100
DO 36,L=1,J	17200
36 IF=IF+MM(IN+L+1)	17300
MM(IN+J+2)=N-M-IF+J+1	17400
J2=J+2	17500
DO 37,L=J2,M	17600
37 MM(IN+L+1)=1	17700
GO TO 42	17800
38 CONTINUE	17900
39 IF(M-1.EQ.J)40,41	18000
40 MM(IN+M+1)=NN-IL	18100
41 CONTINUE	18200
42 DO 43,K=1,M	18300
43 MM(IM+K+1)=MM(IM+K)+MM(IN+K+1)	18400
DO 44,IX=1,M	18500
44 MM(IX)=MM(IM+IX+1)	18600
IF(IJK.EQ.1)45,46	18700
45 IF(NC.LT.10E4) LL=5	18800
IF(NC.LT.10E3) LL=4	18900
IF(NC.LT.10E2) LL=3	19000
IF(NC.LT.10E1) LL=2	19100
IF(NC.LT.10E0) LL=1	19200
GO TO 47	19300
46 IF(NN.LT.10E4) LL=5	19400

47 IF(N,LT,10E4) JJ=5	19500
IF(N,LT,10E3) JJ=4	19600
IF(N,LT,10E2) JJ=3	19700
IF(N,LT,10E1) JJ=2	19800
IF(N,LT,10E0) JJ=1	19900
JKL=5+LL	20000
KK=(130-LL-5)/(JJ+1)	20100
ENCODE(37,48,JSPEC)LL,KK,JJ,JKL,KK,JJ	20200
48 FORMAT(5H(3X,I,I1,4H,2X,I2,2H(I,I1,6H,1X)/(,	20300
C I2,2HX,,I2,2H(I,I1,6H,1X)))	20400
PRINT JSPEC,NN,(MM(J),J=1,M)	20500
49 CONTINUE	20600
END	20700

6.0 COMPARISON

No routine was available to the author for comparison.

7.0 TEST METHOD AND RESULTS

A test program that demonstrates all combinations of 7 "things" taken 4 at a time and specific combinations in accordance with the established correspondence of the individual combinations and the positive integers along with the subroutine printouts follow.

```
PROGRAM TCMBN
DIMENSION II(20), KARRAY(5)
DATA (KARRAY=3,12,7,20,5)
N=7
M=4
CALL CMBN(N,M,0,II)
PRINT 1
1 FORMAT(10(/))
DO 2 I=1,5
CALL CMBN(N,M,KARRAY(I),II)
2 CONTINUE
END
```


(Page 14 is Blank)

1	1	2	3	4
2	1	2	3	5
3	1	2	3	6
4	1	2	3	7
5	1	2	4	5
6	1	2	4	6
7	1	2	4	7
8	1	2	5	6
9	1	2	5	7
10	1	2	6	7
11	1	3	4	5
12	1	3	4	6
13	1	3	4	7
14	1	3	5	6
15	1	3	5	7
16	1	3	6	7
17	1	4	5	6
18	1	4	5	7
19	1	4	6	7
20	1	5	6	7
21	2	3	4	5
22	2	3	4	6
23	2	3	4	7
24	2	3	5	6
25	2	3	5	7
26	2	3	6	7
27	2	4	5	6
28	2	4	5	7
29	2	4	6	7
30	2	5	6	7
31	3	4	5	6
32	3	4	5	7
33	3	4	6	7
34	3	5	6	7
35	4	5	6	7

3	1	2	3	6
12	1	3	4	6
7	1	2	4	7
20	1	5	6	7
5	1	2	4	5

Unclassified

Security Classification

UNCLASSIFIED

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

Naval Research Laboratory
Washington, D. C. 20390

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3. REPORT TITLE

A FORTRAN SUBROUTINE TO PRODUCE COMBINATIONS OF n DISTINCT
THINGS TAKEN m AT A TIME

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

A final report on this phase of the problem.

5. AUTHOR(S) (First name, middle initial, last name)

James P. Grimes

6. REPORT DATE

July 1971

7a. TOTAL NO. OF PAGES

18

7b. NO. OF REFS

1

8a. CONTRACT OR GRANT NO.

NRL General and Administrative
b. PROJECT NO. Function 78-1601

9a. ORIGINATOR'S REPORT NUMBER(S)

NRL Memorandum Report 2340

c.

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned
this report)

d.

NRL Computer Bulletin 25

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

Department of the Navy
(Office of Naval Research),
Washington, D.C. 20360

13. ABSTRACT

A Fortran subroutine to generate all or particular combinations of n things taken m at a time is herein described. The procedure establishes and tabulates a certain one-to-one correspondence between the integers from 1 to C_m^n and the individual combinations of n distinct "things" n_m taken m at a time, where

$${}_nC_m = \frac{n!}{m!(n-m)!}.$$

In all cases, the set of "things" treated by the subroutine is the set of integers $\{1, \dots, n\}$.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Combinations Computer Subroutine Statistical Analysis Probability One-to-One Correspondence Mathematics Parameters						

UNCLASSIFIED

